#### SPL-1 Project Report, 2020

**Data Classification using Decision Tree**

#### SE 305 : Software Project Lab I

Submitted by

**Islam Bin Md Junaed**

BSSE Roll No. : **1134**

Supervised by

**Dr. Md. Shariful Islam**

Designation: Professor

Institute of Information Technology



Institute of Information Technology University of Dhaka

24-08-2021

## Table of Contents

1. [Introduction 3](#_heading=h.gjdgxs)
   1. [Background Study 3](#_heading=h.30j0zll)
   2. [Challenges 3/4](#_heading=h.1fob9te)
2. [Project Overview 4/5](#_heading=h.3znysh7)
3. [User Manual 5/6](#_heading=h.2et92p0)
4. [Conclusion](#_heading=h.tyjcwt) 7
5. [Appendix](#_heading=h.3dy6vkm) 7

[References](#_heading=h.1t3h5sf) 7

# Introduction 3

On this project, I’ll be implementing a decision tree using Python language. Decision tree is the most powerful and popular tool for classification and prediction. A Decision tree is a flowchart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.

# Background Study

# Python was a new language to me so I had to get acquainted with Python at the beginning of this project which was the first phase of Background Studies. Then I learned about the basic concepts of Machine Learning.

# As my project is about the Decision Tree algorithm implementation, and since the Decision Tree is a Classification Algorithm, I had to learn about Machine learning, Data Classification’s basic from beginning.

# Then I learned the Algorithm.

# But, I wasn’t able to implement the algorithm because I was new to all of these.

# So, I started following a video series non YouTube where the content creator

# Implemented a step by step Decision Tree algorithm and converted it into coding- The

# Link of which I have given in the reference section. Although I’m not really good at

# anything, I’ve enjoyed working with Python, Data Classification, Machine Learning, Data

# sets- all of these interesting topics for the first time. Hopefully, these topics will help me

# to learn something better in future.

# Challenges

There are a number of challenges in implementing a new software solution. The process can be confusing and overwhelming. There are many challenges I have faced to implement this project. There are some of them –

* + - Working with csv file for the first time
    - This algorithm is totally a new concept to me
    - Some new learning –
      * Entropy
      * Information Gain
      * Recall
      * Precision
    - Running a recursive function to build the decision tree
    - finding the best split feature and best split value on each level of the tree
    - Linking the tree nodes - parents node to child node
    - Learning the pandas library and numpy library were not easy for me. 4

# Project Overview

A **decision tree** is a decision support tool that uses a [tree-like](https://en.wikipedia.org/wiki/Tree_(graph_theory)) model of decisions and their possible consequences, including [chance](https://en.wikipedia.org/wiki/Probability) event outcomes, resource costs, and utility. It is one way to display an algorithm that only contains conditional control statements.

Decision trees are commonly used in operations research, specifically in decision analysis, to help identify a strategy most likely to reach a goal, but are also a popular tool in machinelearning.

Here I have implemented a decision tree algorithm on the Iris flower dataset. The Iris dataset contains information about three species of Iris flowers -

* Iris Setosa
* Iris Virginica
* Iris Versicolour

We have four features and depending on these features we can classify the Iris flower among the three species. The features are-

* Sepal Length in cm
* Sepal width in cm
* Petal length in cm
* Petal width in cm

Based on the features we have to make a decision whether a given flower is Iris Setosa or Iris Virginica or Iris versicolour.

The making of decision tree consists of the following steps:

**2.1. Preparing dataset**

We have used the Iris.csv file from the Kaggle dataset. We have read the dataset and split it into training and test dataset. We have randomly selected a certain size of test dataset and the rest is used as a training dataset.

**2.2 Build decision tree**

Now we have to work on the training dataset. We have to select the best split feature and the best split value as the decision tree root. For this we will calculate **entropy** and **information gain** and then we will choose the feature and value with the most information gain value i.e. least entropy value.

Entropy = sum(-probability\* log2(probability))

After selecting the root we have two kinds of decisions: yes and no. For the yes answer we will head through the right sub-tree and for the no answer we will head through the left sub tree. We will recursively run the algorithm to find the left sub-tree and the right sub-tree until we reach our maximum depth of the tree or we find a pure value which is our result.

5

**2.3 Testing the decision tree**

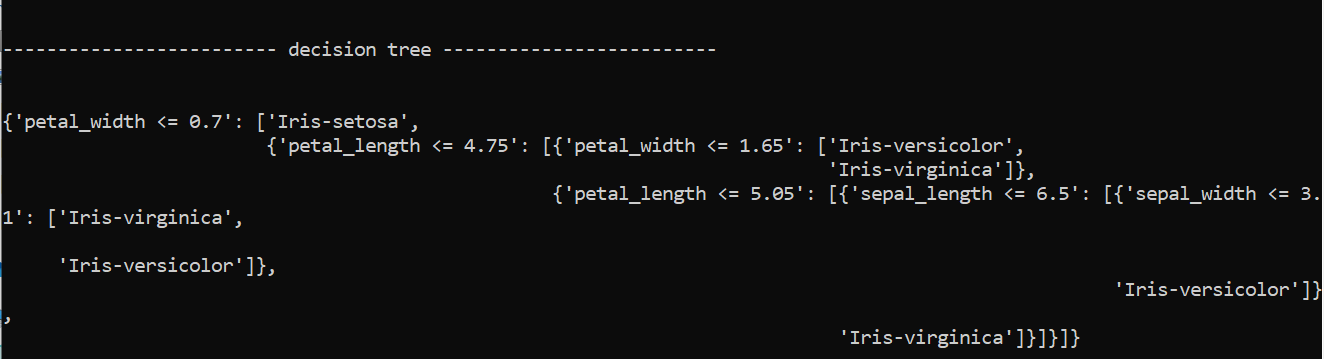
After building the decision tree with the training dataset we will now test it with our test dataset. For the features of the test dataset we will predict the flower species and match with the actual value. This way we will know how accurate our decision tree is. We have calculated accuracy, recall, precision and f-measure of the tree to measure how reliable our decision tree is.

# User Manual

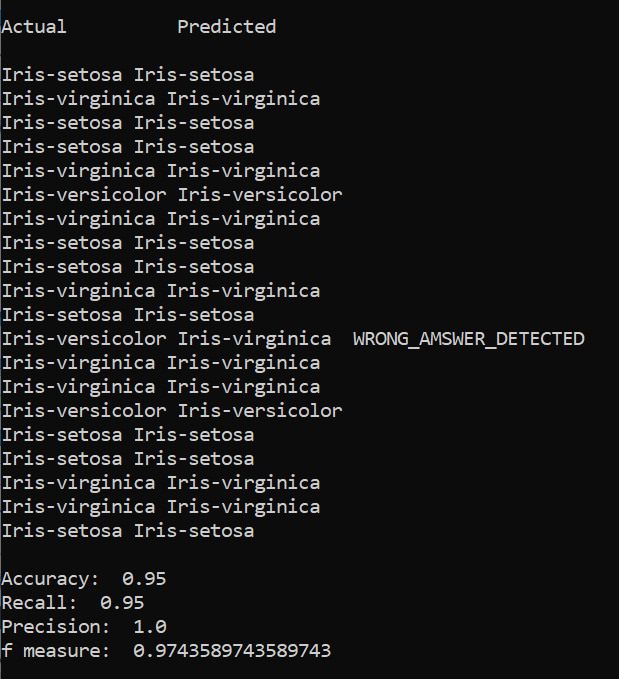
Users just need to run the program. Before that they need to make sure that the file is placed in the same directory as the DecisionTree.py file. So type command in the cmd -

-> python DecisionTree.py

And we will get the output as following -



Here maximum depth was defined 5 and we can see a decision tree of height 5. And we have also tested this tree with our extracted test dataset. The resting result is as following-



Under the actual column we can see the values of the actual dataset. And under the predicted column we can see the values that we predicted using our own built decision tree. We can also see tha among 20 test sets only 1 test case our decision tree failed to predict the right answer.

After this we have shown the decision tree’s accuracy, recall, precision and f-measure.

# Conclusion 7

This project let me enter the realms of machine learning for the first time. Implementing the “Decision Tree” algorithm was fun and challenging at the same time. In a word, it has been an adventure to learn how statistics work with probability, inference and machine learning. I also learned how to calculate the entropy, logarithmic functions more precisely. It was a great experience and I can not be more thankful to my supervisor who constantly helped me wherever I struggled and led my path. I really look forward to using this experience in future projects and programming works.

# Appendix

I got familiar with some other advanced machine learning algorithms which I eagerly look forward to working on in the future.

# References

###### https://www.youtube.com/watch?v=y6DmpG\_PtN0&feature=youtu.be

###### <https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/>

###### 3. <https://www.sebastian-mantey.com/code-blog/coding-a-decision-tree-from-scratch-python-p1-introduction>

###### <https://www.geeksforgeeks.org/decision-tree/>

###### Thank You